ECE 321 - Introduction to Software Engineering
Homework 2

Exercise 1

Write a program in `C` which calculates whether a number is perfect or not using three methods:

1. A `for` loop
2. A `while` loop
3. A `do-while` loop

In number theory, a perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself. The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and $1 + 2 + 3 = 6$. The next perfect number is $28 = 1 + 2 + 4 + 7 + 14$.

Exercise 2

Write a program that reads two real numbers X and Y and answers where the point with coordinates (X, Y) is in a circle with radius 1.
Input example 1:

```
0.5 0.5
```

Output example 1:

```
inside
```

Input example 2:

```
-1 0
```

Output example 2:

```
border
```

Input example 3:

```
2 -3.5
```

Output example 2:

```
outside
```

Exercise 3

Write a program that reads two natural numbers, A and B, and finds if one of the two is divisible by 17 and the other is divisible by 42 at the same time. The numbers should be within specific limits:

- $0 \leq A$
- $B < 1,000,000$

Input example 1:

```
210 34
```

Output example 1:

```
yes
```

Input example 2:

```
1717 4224
```

Output example 2:

```
no
```

General Software Development Standards and Guidelines

Windows users: *In order to use the functions as they were implemented in libc, you* ***may*** *have to put the following line as the first line of your code: (if you see an error message about function unsafety)*

```
#define _CRT_SECURE_NO_WARNINGS
```

Submissions must include only the source files (.c/.cpp) files. Also, each file must have an appropriate name e.g. `hw1-ex1.c`

Programming assignments will be evaluated based upon fulfillment of functional requirements, coding style and documentation.

Internal documentation improves the readability of a software module. Many of the general software development guidelines are focused on using good internal documentation practices. A file containing one or more software modules or a shell script file should have a comment block at its beginning containing the following basic information:

- The name of the author who created the file
- The date the file was created
- Description (overview of the purpose of the modules)
- A statement that declares who contributed in the development of the code
- Any other information that you believe it is necessary

General coding standards pertain to how the developer writes code:

Indentation: Proper and consistent indentation is important in producing easy to read and maintainable programs. Indentation should be used to:

- Emphasize the body of a control statement such as a loop or a select statement
- Emphasize the body of a conditional statement
- Emphasize a new scope block

A minimum of 3 spaces shall be used to indent. Generally, indenting by three or four spaces is considered to be adequate. Once the programmer chooses the number of spaces to indent by, then it is important that this indentation amount be consistently applied throughout the program.

Structured Programming: Structured (or modular) programming techniques shall be used. GO TO statements shall not be used as they lead to code, which is hard to read and maintain.

Variable Names: Variables shall have mnemonic or meaningful names that convey to a casual observer, the intent of its use.